



# **Rapport Bursdagstjenesten**

## **Mappe 2: Apputvikling**

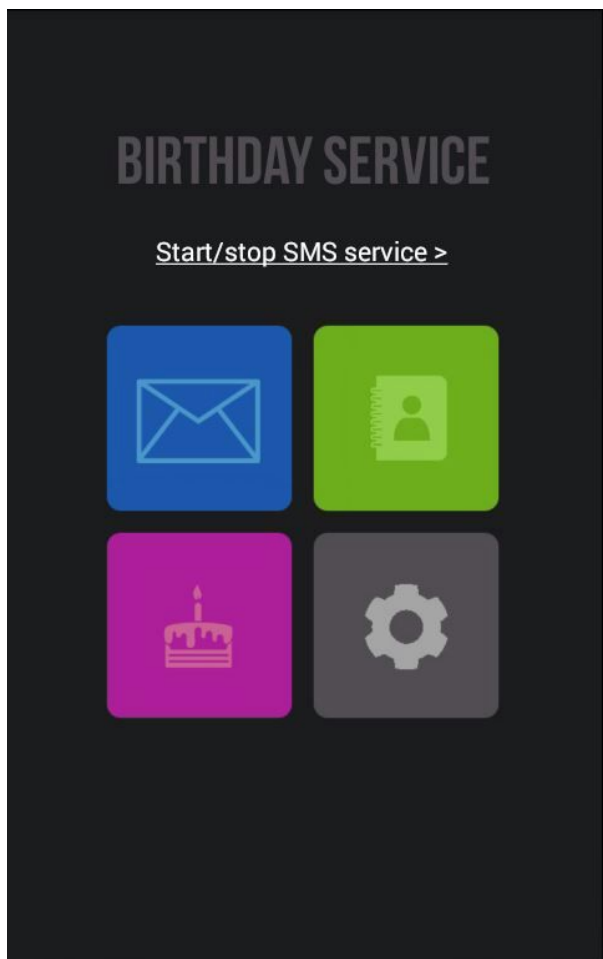
### **Høst 2015**

Av Annika Hammervoll (s198611)  
og Helene Juterud (s198541)

## 1. Innledning

I andre mappe-oppgave skulle vi lage en applikasjon som sender ut melding til kontakter på deres bursdag. I applikasjonen skulle det være mulig å legge til/fjerne/endre kontakter, samt redigere meldingen som skal sendes og tidspunktet for utsendelse.

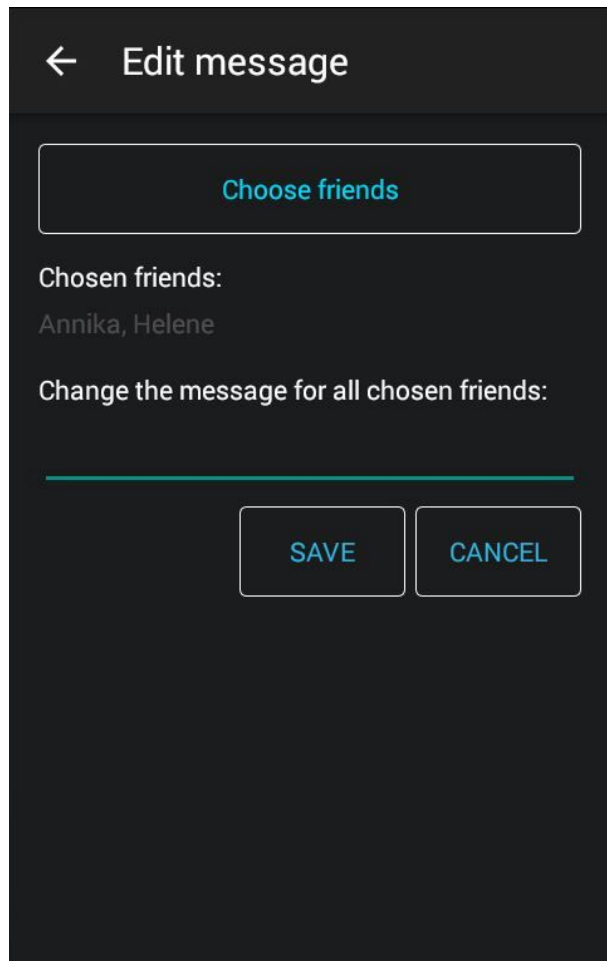
## 2. Design og navigering



Ved oppstart av applikasjonen kommer man til en hovedmeny, som består av 4 knapper. Dette er for å gi brukeren bedre oversikt og en følelse av at man er på starten av en applikasjon. Alle knappene har ulik farge og ikon for å lettere kunne skille de ulike alternativene fra hverandre. I utgangspunktet skulle innstillinger-knappen være gul for å stå i stil med de andre. Vi gikk bort fra dette, og endret fargen til grå, da vi så at gulfargen tok mye av fokuset vekk fra hovedfunksjonene. Mange av aktivitetene inneholder mye informasjon eller felter som brukeren skal forholde seg til, og for å begrense mengden støy valgte vi å fjerne drop-down (options) menyen som automatisk implementeres i actionbaren. Hovedmenyen inneholder snarvei til alle metodene i applikasjonen, og vi mener derfor at en drop-down meny ville vært overflødig her.

Under planleggingen av applikasjonen fant vi inspirasjon i melding- og kontakter-tjenesten som følger med på android-telefoner. Ved å gjenbruke noe av strukturen (plassering av

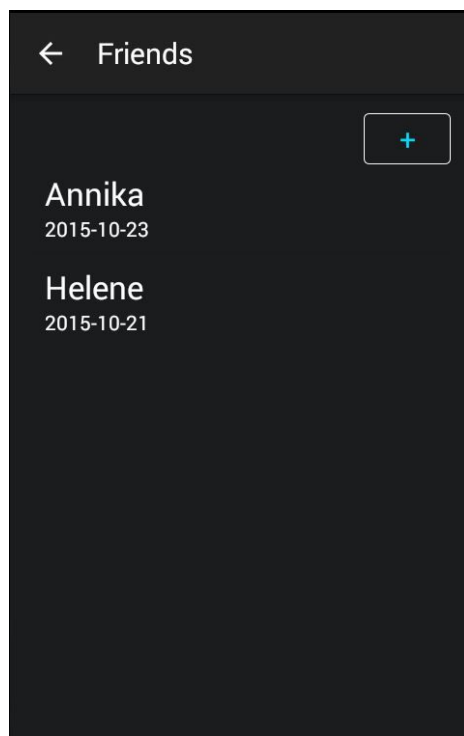
knapper etc) håper vi at brukere vil gjenkjenne funksjonalitet og ha økt forståelse av applikasjonen, allerede ved første bruk.



#### Melding-siden:

Inne på meldingsiden (blå ikon) får man muligheten til å endre meldingen som skal sendes til en bestemt kontakt eller flere.

Når man klikker på "Choose friends" får man opp et listview der alle kontaktene lagret i databasen blir listet opp. Man kan da sjekke av de man ønsker å endre meldingen til. Etter å ha klikket ok, kommer man tilbake til Edit Message-siden, og navnene på de valgte vennene vises under Chosen friends. Deretter er det bare å skrive inn den nye meldingen og klikke save. Det er lagt inn validering av input slik at ikke uønskede verdier lagres i databasen.



#### Kontakter-siden:

Når man trykker på den grønne knappen i hovedmenyen kommer man til kontakter-siden. Her blir alle kontakter hentet ut fra databasen og vist med navn og bursdagsdato. Man har muligheten til å legge til en ny kontakt, eller å redigere en kontakt herfra.

I motsetning til choose friends-siden, er + knappen på denne siden liten, og strekker seg ikke over hele bredden. Det kan kanskje være litt forvirrende at knappene har forskjellig form, men vi valgte å ha det slik for å unngå at forskjellig funksjonalitet skal ha samme form. På kontakter-siden er legg til kontakt-funksjonen ikke den eneste funksjonen på siden. Mens på velg kontakt-siden knyttet til Edit Message er hovedpoenget å velge kontakter og deretter bare trykke ok. Ok-knappen er derfor viktigere i det skjermbildet, og derfor større.

← Edit friend

Name:  
Annika

Phone number:  
12345679

Birthday:  
Sep 22 2014  
Oct 23 2015  
Nov 24 2016

Message:  
Happy Birthday!

SAVE DELETE CANCEL

#### Endre kontakt:

Ved å klikke på en av kontaktene som er listet opp inne på kontakter-siden, åpner man vinduet man ser til venstre, og all informasjonen som er lagret i databasen om kontakten vil være fylt ut. Her kan man endre eller fjerne den valgte kontakten. Hvis man trykker på cancel-knappen kommer man tilbake til kontakter-listen, mens hvis man trykker på tilbake-knappen øverst i venstre hjørne kommer man tilbake til hovedmenyen. Det er lagt inn validering av input når man forsøker å lagre endringene og det gis tilbakemeldinger til brukeren alt etter hva denne foretar seg.

← Add a new friend

Name:

Phone number:

Birthday:  
Sep 22 2014  
Oct 23 2015  
Nov 24 2016

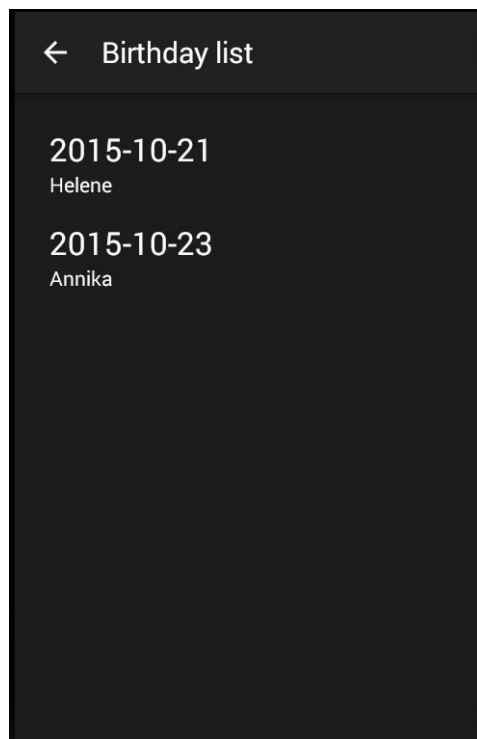
Message:  
Happy Birthday!

ADD CANCEL

#### Legg til ny kontakt:

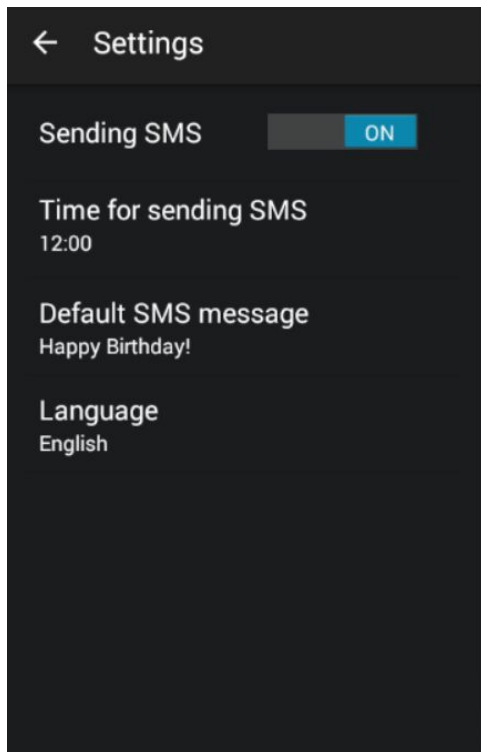
Hvis man velger å trykke på + knappen på kontakter-siden (i motsetning til å velge en bestemt kontakt), får man muligheten til å legge til en ny kontakt/venn. Da får man opp vinduet man ser til venstre. Her legger man inn navn, telefonnummer, velger bursdagsdato og skriver inn meldingen man vil sende. Når man åpner denne siden vil default/standard-meldingen (i dette tilfellet "Happy Birthday!") allerede stå i meldingsfeltet. Brukeren kan endre denne standard-meldingen på innstillinger-siden. Dersom man ønsker å skrive en helt spesiell melding til denne kontakten, kan man bare skrive inn sin egen melding i meldingsfeltet her når man legger til en ny venn. Hvis man trykker på cancel-knappen kommer man tilbake til kontakter-oversikten, mens hvis man trykker på tilbake-pilen øverst i venstre hjørne kommer man tilbake til hovedmenyen. Det er selvsagt lagt inn validering av input også på denne siden og brukeren

får tilbakemelding på det denne foretar seg.



#### Bursdagsliste:

Klikker man på det rosa ikonet på forsiden, kommer man inn på bursdagslisten. Dette er kun et vindu som viser alle registrerte kontakter, sortert på bursdagen sin. Vi syntes dette var en grei funksjonalitet siden brukeren da kan få en litt bedre oversikt over bursdagene.



### Innstillinger:

Innstillinger-siden får man opp ved å klikke på det grå tannhjul-ikonet på forsiden. Her kan man skru meldingstjenesten av og på, endre tidspunktet man vil sende ut SMS på, endre standard-meldingen, og endre språket. Ved å klikke på Language endres språket til norsk, ved å klikke på Default SMS message kommer det opp en dialogboks hvor man kan sette endringen eller avbryte, ved å klikke på Time for sending SMS kommer det opp en dialogboks med en timepicker hvor man kan sette endringen eller avbryte, og Sending SMS-tjenesten styres av en switch som kan være av eller på. For at man første gang man benytter applikasjonen skal vite at man må starte tjenesten, har vi lagt inn en liten link under overskriften på hovedsiden som sier Start/stopp SMS-tjenesten >. Trykker man på denne kommer man hit. Når brukeren endrer tidspunktet for utsendelse har vi dessuten lagt inn at switchen skrus av siden tidspunktet må

oppdateres i tjenesten. På den måten startes tjenesten opp igjen med det nye tidspunktet når brukeren slår switchen på igjen.

### 3. Oppbygging av kode

Vår applikasjon består av 20 klasser, der ca. 10 av disse har tilhørende XML-filer.

BirthdayServiceMainActivity: Denne klassen er som navnet sier vår hovedaktivitet. Det er i denne aktiviteten hovedmenyen blir opprettet og vises.

BirthdaysActivity og BirthdaysFragment: Dette er bursdagslisten vår som man kommer til ved å trykke på det rosa bursdagskake-ikonet på forsiden. Her benyttes Androids innebygde `simple_list_item_2`, som viser bursdagen øverst i størst skrift og navnet til personen under i litt mindre skrift. Vi bruker Content Provideren vi har opprettet for prosjektet vårt for å hente ut alle kontaktene fra databasen og sortere på bursdag. Det beste her hadde vært å ignorere årstallet og sortere først på måned (som et tall) og så på dag (som et tall), men dette fant vi dessverre ikke ut av før innleveringsfristen. Bursdagen til hver person i databasen er lagret som en string og dette skapte problemer akkurat her, men gjorde det lettere andre steder i applikasjonen.

ChangeMessageForContactsActivity, ChooseContactsActivity og ChooseContactsFragment: `ChangeMessageForContactsActivity` er aktiviteten som kjøres når man velger den blå melding-knappen i hovedmenyen, og man kommer til siden for å endre meldingen til en eller flere kontakter. Denne klassen inneholder kode for å gå til `ChooseContactsActivity` når man trykker på "Choose friends"-knappen og å fylle inn en liste med `Person`-objekter ut fra hvilke personer som har blitt valgt i `ChooseContactsActivity` og `ChooseContactsFragment`. Her er det også lagt inn validering (regEx) av meldingsfeltet og en sjekk på at man faktisk har valgt noen kontakter når man trykker på lagre-knappen. `ChooseContactsActivity` er klassen som henter inn `ChooseContactsFragment`, og `ChooseContactsFragment` henter på samme måte som `BirthdaysFragment` ut alle personene fra databasen. I `ChooseContactsFragment` har vi valgt å bruke Androids innebygde `simple_list_item_multiple_choice` for å få checkbox'er for hver kontakt man ønsker å endre meldingen til. Her er det bare navnet til kontakten som vises sammen med checkbox'en. Når en kontakt velges, registreres det hvilken id denne kontakten har i en `ArrayList`, og denne id'en fjernes også fra denne `ArrayList`'en hvis brukeren angrer og un-checker kontakten. Informasjonen om alle id'ene som har blitt valgt sendes tilbake til `ChangeMessageForContactsActivity` når brukeren trykker på ok-knappen (gjennom en `Intent`), og det er disse id'ene man går ut fra når listen med `Person`-objekter fylles i `ChangeMessageForContactsActivity`.

DBHandler: Denne klassen extender `SQLiteOpenHelper` og inneholder alle database-metodene.

Person: Dette er en hjelpeklasse som korresponderer med database-tabellen `Person`, som opprettes i `DBHandler`.

PersonCP: Dette er Content Provideren til prosjektet.

ContactsActivity og ContactsFragment: `ContactsActivity` er aktiviteten man kommer til når man trykker på det grønne kontakter-ikonet på forsiden. Denne inneholder fragmentet

ContactsFragment, som benytter Androids innebygde simple\_list\_item\_2 og viser navnet til kontakten øverst og bursdagen til kontakten under. I ContactsFragment ligger også kode for hva som skal skje når brukeren trykker på en av kontaktene: Id'en til denne personen brukes til å hente inn den valgte personen fra databasen, og alle feltene til denne personen legges med en intent som sender brukeren til EditContactActivity. Hvis brukeren trykker på + knappen så sendes brukeren videre fra ContactsActivity til NewContactActivity.

EditContactActivity: Dette er aktiviteten man kommer til hvis man velger (trykker på) en av kontaktene i kontakter-listen (ContactsActivity og ContactsFragment). Her henter man inn informasjonen til personen fra Intent'en og fyller inn de ulike feltene med denne. Man kan endre alle feltene, og hvis man trykker på lagre-knappen valideres input'ene og år, måned og dag hentes ut fra DatePicker'en. Hvis alt er i orden lagres dataene i databasen og det kommer opp en Toast med informasjon om at informasjonen til denne kontakten har blitt endret før man går tilbake til kontakter-listen, som oppdateres med informasjonen til den nye kontakten. For at kontakter-listen nå skal oppdateres måtte vi velge å sende en Intent til ContactsActivity og dette er årsaken til at man kommer til hovedmenyen når man trykker på tilbake-knappen øverst i venstre hjørne - vi måtte finish'e den forrige ContactsActivity siden vi her i EditContactActivity på oppdatere den (og dermed sende en ny Intent). Hvis man trykker på avbryt-knappen kommer man tilbake til ContactsActivity med kontakter-listen.

NewContactActivity: Denne aktiviteten kommer man til når man trykker på + knappen i ContactsActivity (kontakter-listen). Her henter man inn default-meldingen (som brukeren kan endre på innstillinger-siden) og plasserer den i meldingsfeltet. Brukeren kan avbryte ved å trykke på avbryt-knappen, og kommer da tilbake til kontakter-listen som forblir uendret, eller brukeren kan fylle inn feltene og registrere en ny kontakt ved å trykke på legg til-knappen. Hvis man trykker på add/legg til valideres input-feltene, år, måned og dag hentes inn fra DatePicker'en, og hvis alt er i orden lagres den nye kontakten i databasen og det dukker opp en Toast som sier at personen har blitt lagt til. Da kommer man også tilbake til den oppdaterte kontakter-listen. Hvis noen av input-feltene ikke validerer, benytter vi setError-metoden for hvert aktuelle felt og legger inn en forklarende melding på hva som er galt. Denne metoden for å gi tilbakemelding til brukeren om feil benyttes også i EditContactActivity.

MyBroadcastReceiver, SetService, MyService og ResultOfNotificationActivity: Dette er klassene som har med SMS-tjenesten å gjøre. Når switchen på innstillinger-siden endres (skrur på eller av), sendes det ut en broadcast som MyBroadcastReceiver snapper opp. Denne sender så en Intent videre til SetService, som sjekker om switchen i innstillinger er av eller på. Hvis den er på hentes tidspunktet for utsendelse av SMS ut (satt i innstillinger) og det settes en PendingIntent med en Intent til MyService som skal kjøre på dette tidspunktet hver dag - en AlarmManager brukes for å trigge dette. Hvis switchen er av, avbrytes AlarmManager tidligere i SetService. MyService dobbeltsjekker at switchen er skrudd på, og hvis den er det så henter den ut dagen i dag og sammenlikner denne med bursdagsdatoene i databasen. Hvis den finner noen som har bursdag, sendes en SMS til disse med meldingen som brukeren har lagret på disse personene, og brukeren får også en notification øverst på mobilen sin når en SMS har blitt sendt ut. Hvis brukeren trykker på denne notification'en så kommer han eller hun til ResultOfNotificationActivity, hvor det står oppgitt hvem man har



sendt SMS til og telefonnummeret til denne personen, når denne personen har bursdag, og meldingen som har blitt sendt.

SettingsActivity: Dette er aktiviteten man kommer til når man trykker på det grå tannhjul-ikonet på forsiden. Denne aktiviteten inneholder en klasse som extend'er PreferenceFragment, noe som gjør at informasjonen lagres i systemet. Denne klassen og denne siden har blitt omtalt tidligere.

TimePickerDialog: Dette er en klasse som extend'er DialogPreference og utgjør TimePickeren som vises i dialogvinduet når man trykker på Tidspunkt for SMS-utsendelse på innstillinger-siden.

WarningDialog: Dette er en klasse som extend'er DialogFragment og denne dukker opp når man velger å slette en kontakt inne på EditContactActivity. Det er viktig å ha en ekstra sikkerhet når det gjelder sletting for at det ikke skal oppstå feil.

## 4. Forbedringer

Planen var at sidene der man kunne velge mellom kontakter skulle ha en søkefunksjon. Denne fikk vi dessverre aldri tid til å implementere skikkelig, og valgte derfor å fjerne den.

Bursdagslisten kunne vært endret slik at den tok inn dagens dato og sorterte kontaktene etter hvem som har først oppkommende bursdag. Selve sorteringen burde dessuten fikses ordentlig.